〈論　　文〉

トランスピュータを用いたコンピュータネットワークシステム

大　森　義　行

1．はじめに

近年，通信技術の急速な発達と相まって，コンピュータを用いた協調活動支援技術「グループウェア」[1]に対する気運が高まっている。ここでは，共通の目的を持って活動するグループの個々人に対し，スムーズにコミュニケーションできる電子的環境を提供し，仕事の質的な変化をもたらしてきている。

例えば，グループ内の個々人に均等に仕事を分担させるとすれば，処理能力の差異により人によっては遊び時間が生ずるだろうし，また喩え能力を事前に申告していたとしても，課題によっては予め完了時間を見積もれないケースも考えられる。この場合，実社会のような見張り役を設け，処理度合いに応じ仕事を分配していくことが肝要である。

この様に監視役も含め全ての機構をコンピュータで実現させる手法の一つにプロセッサファーム (Processor　Farm) 技法[2]がある。これは監視役に当たるコントローラと処理を行うワーカから構成され，一般にコントローラはワーカの処理状況を監視し，ワーカはコントローラから仕事を得て処理し結果を帰す役割を果たすこととなる。プロセッサファームを実現するために必要となるものとしてはコンピュータ（プロセッサ）通信機能のみである。従って，どのようなアーキテクチャやネットワーク上でもファームシステムを構築することは可能であるが，LANで接続された実際のコンピュータシステムを利用するためには，UNIXやWindowsで提供される通信機能に関する相当な知識が要求される。

本報告では，プロセッサ間通信を容易かつ高速に実現できるマイクロプロセッサ，トランスピュータ (Transputer)[3]を用い，マルチプロセッサによるファームシステムを構成し，プロセッサの個数変化に対するネットワークシステムの挙動および処理効率などを求めシステムの特性を評価した。また，ネットワークを構築するにあたり考慮しなければならない事項などについて考察を行った。

2．ネットワークシステム

2．1　トランスピュータ[3]

トランスピュータは，並行プロセスが互いにデータ通信しながら処理を効率よく実現するために，CSP (Communicating Sequential Processes) の概念に基づき設計された32ビットRISCタイプのマイクロプロセッサである。単体における性能はメーカーの公称値で12.5MIPS，1.87MFLOPS (T800-25MHz) であり，パソコン (i486) あるいはワークステーション (SPARC) に比較しても高い処理能力を有している。

トランスピュータとは，TRANSistor-comPUTERという造語と言われ[3]，「トランジスタのように何個でも接続することができるコンピュータ」という意味合いを持つ。トランスピュータ（T800）は，4kbyteの内部RAM，メモリインターフェース，他のトランスピュータと通信するため入出力の信号線2

本で1組となった4個のシリアルリンクを所有しており，トランスピュータ同士のリンクを接続することにより容易にマルチプロセッサネットワークシステムの構築が可能となる。また，リンクの最大通信速度は20Mbpsであり，非常に高速な通信が可能となっている。

トランスピュータはボード上に構成されており，トランスピュータボードを接続するパソコンなどの開発マシンをHostコンピュータ，これに直接接続されるトランスピュータをRootトランスピュータと呼ぶ。なお，トランスピュータではディスプレイやキーボード等の入出力装置を所持していないため，Hostコンピュータを介して入出力が行われなければならない。このためHostコンピュータにAfserverおよびRootトランスピュータにはFilterという入出力関係のタスクを配置しなければならない。

2．2 パラレルFORTRAN[4]

トランスピュータは元来，並列処理言語OCCAMのために開発されたマイクロプロセッサであるが，既存のソースプログラムを活用するためにCおよびFORTRANが提供されている。パラレルFORTRANではチャンネル通信によるマルチタスク処理が可能であり，そのためのチャンネル関数が用意されている。チャンネル関数を用いた例として，「あるチャンネルから配列データを入力し，そのまま別のチャンネルへ出力する」というサンプルプログラムを図1に示す。

ここではデータ通信のため，F77_CHAN_IN_MESSAGE，F77_CHAN_OUT_MESSAGEというチャンネル関数を使用している。両関数とも，第1引数は入出力するバイト数，第2引数はデータを格納する変数，第3引数はチャンネルアドレスである。バイト数としては，送受信するデータの合計を指示しなければならず，ここでは，単精度，要素数10の配列であるから40バイトとしている。

トランスピュータを用いパラレルFORTRANのプログラムを実行するためには，ハードウェアおよびソフトウェアに関する接続情報を記述したコンフィグレーションファイルを作成しなければならない。例えば，図2のようにRootトランスピュータのみを使いMainタスクを実行しようとするコンフィグレーションファイルは図3のようになる。

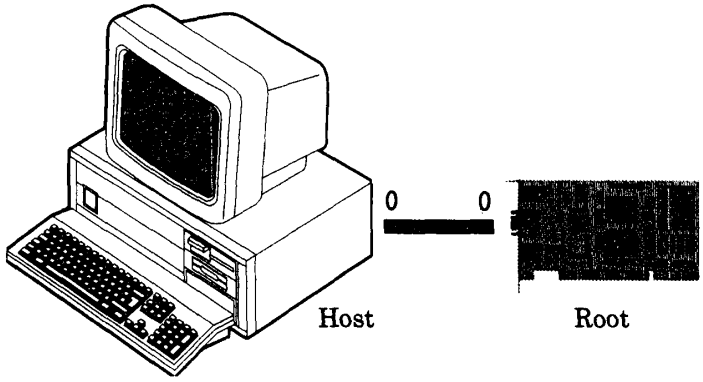
ここで①トランスピュータの使用宣言，②各トランスピュータにおけるリンクの物理的接続情報，③各トランスピュータで実行されるタスク名と使用チャンネル，④タスクの配置，⑤タスク間の通信チャンネルの情報，を表している。

図1 チャンネル関数を利用したサンプルプログラム

```
INCLUDE 'CHAN.INC'
INTEGER INCHAN,OUTCHAN
DIMENSION A(10)
INCHAN=F77_CHAN_IN_PORT(0)
OUTCHAN=F77_CHAN_OUT_PORT(0)
CALL F77_CHAN_IN_MESSAGE(40, A, INCHAN)
CALL F77_CHAN_OUT_MESSAGE(40, A, OUTCHAN)
END
```

図2　プロセッサ間接続とタスク配置

a）ハードウェア接続



b）ソフトウェア接続

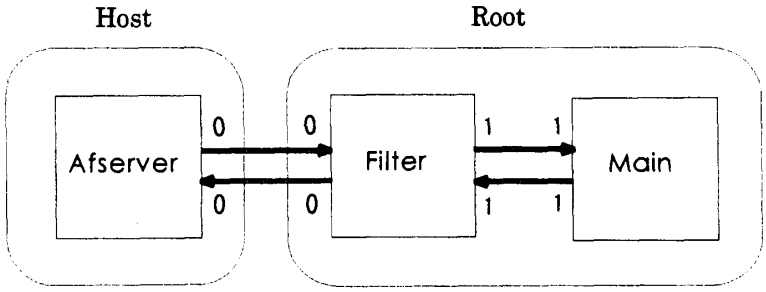
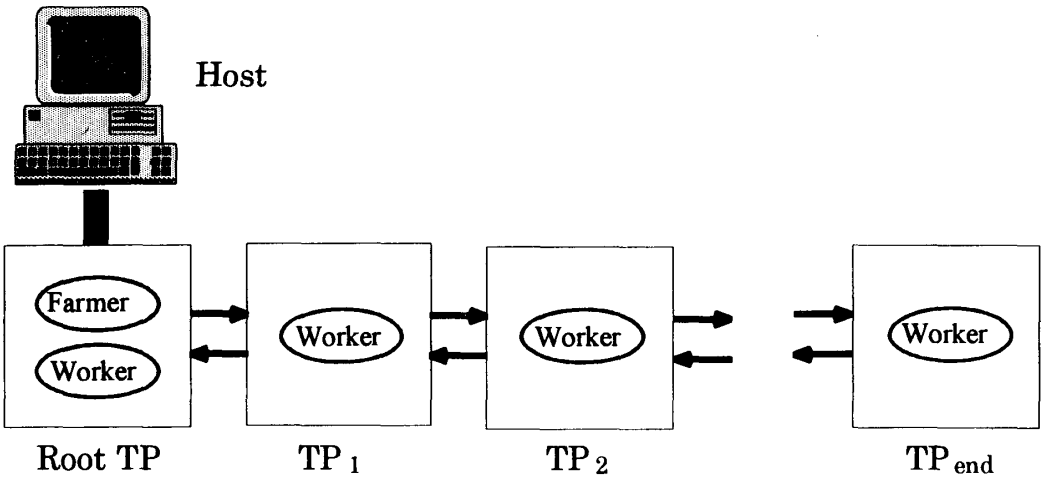


図3コンフィグレーションファイル

```
processor     Host
processor     Root

wire?          Host[0]  Root[0]

task           Afserver Ins=1  Outs=1
task           Filter    Ins=2  Outs=2  Data=10k
task           Main      Ins=1  Outs=1  Data=350k

place          Afserver  Host
place          Filter    Root
place          Main      Root

connect?       Afsever[0]  Filter[0]
connect?       Filter[0]   Afsever[0]
connect?       Filter[1]   Main[1]
connect?       Main[1]     Filter[1]
```

図4 ネットワーク構成



（入出力関係のタスクは除く）

2．3 ネットワークの構成

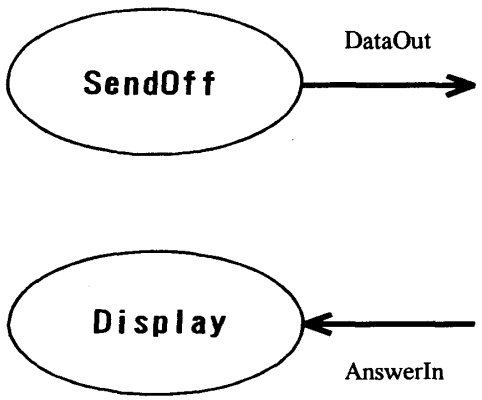
ネットワークの構成にあたり，マルチプロセッサが処理する並列的解法のほとんどの種類に適用でき，いったん基本的なソフトウェアが作られれば簡単に様々な問題に応用できるプロセッサファーム技法[2]を用いた。

プロセッサファーム（以降，ファームと略す）とは，互いに独立したプロセッサが集まったものであり，本報告においてはファームの接続は図4に示すようなパイプライン型で，ルートトランスピュータを含め最大9台まで接続可能である。各プロセッサはコントローラにより割り当てられたタスクを実行し，その結果を帰し，新しい仕事を待つことになる。このときコントローラーは，プロセッサが何らかの仕事をしているbusy状態にあるか，何もしていないidle状態であるか，どの仕事が完了したのか，そしてどの仕事が残っているのか等の状態を常に監視することとなる。従って，ファームを実現するために唯一必要な条件は「プロセッサ間通信」であり，これを高速に実行できるトランスピュータはファームシステムを構成するのに最適なものであると言える。ここでは，各々のトランスピュータにファーマプロセスおよびワーカプロセスを図5のように配置している。

ファーマプロセスは，処理すべきデータを順にワーカプロセスへ送出し，ワーカプロセスで処理された結果を受取り，グラフ表示するとともにファイルに格納している。これは，図5に示すような，SendOffとDisplayの2つのタスクで構成した。また，この2つのタスクは並列に実行されている。

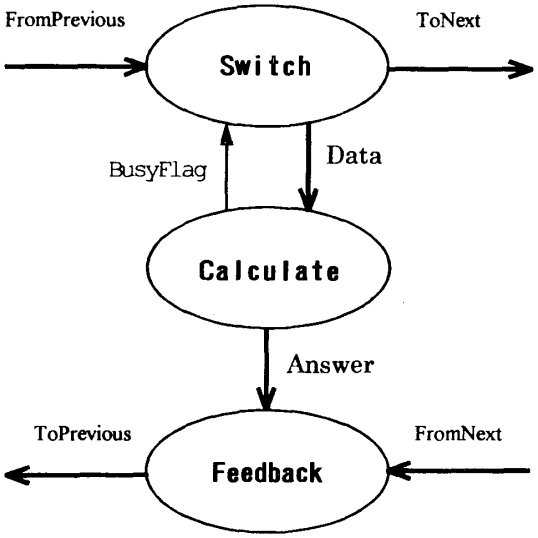
各ワーカプロセスはデータを受取り，idle状態であれば自ら処理を行いファーマプロセスに結果を帰し，busy状態であれば次のワー

図5 farmerプロセスの要素



カへデータを受け渡す。これらは図6に示すSwitch, Calculate, Feedbackの3つのタスクで実現されている。

図6　workerプロセスの要素



Switchタスクでは，送られてきた入力データを一時保管し，Calculateタスクで処理が行われているかを判断するBusyFlagを参照し，TRUEであれば入力データをCalculateタスクへ渡し計算を依頼するとともにBusyFlagをFALSEに換える。一方，FALSEである場合は次のワーカへデータを送り出す。ただし，1番最後のワーカ（TP_{end}）では次のトランスピュータがないためタスク機構は多少異なる。

Calculateタスクは，入力データを受取り計算を行い，出力結果をFeedbackタスクへ送信する。さらに、処理が終了するとBusyFlagをFALSEに換え、Switchタスクへ新しい処理が可能であることを伝える。

Freebackタスクでは、Calculateタスクあるいは次のトランスピュータから送られてきた出力データを、前のトランスピュータもしくはファームプロセスへ送信する。

ネットワークを用いプログラムを実行するためには、上述のタスクを実際のトランスピュータに配置するためのコンフィグレーションファイルの情報に従って各トランスピュータに配置されてネットワークコンピューティングが実行される。

3．実験結果

ネットワークの効率などを検討するためにCalculateタスクに具備すべき条件としては，様々な演算あるいは関数計算を含み，かつ同一のプログラムにおいても解析条件により計算時間が異なることが挙げられる。また，本システムに用いたT800型トランスピュータは数値計算に適していると言われ，Calculateタスクにおいては，非線形の連立微分方程式からなるボルツマン方程式解析[5]を対象とした。この解析では，電界を入力データとし電子群の平均速度などが結果として計算される。この時，与えられる電界の大きさで微分方程式の収束が変わり演算時間が大幅に異なることが知られている。本解析では電界Eの値を10～400（V/cm）まで10間隔に40個のデータに対して計算を行った。

表1は，Rootトランスピュータ1台のみ使用し，CalculateタスクをDOループによる繰り返しで実行する従来通りのシーケンシャル処理と，本報告で開発されたファーマプロセスおよびワーカプロセスを配置したパラレル処理の一部を利用する，2つのケースの処理時間である。

パラレル処理の結果はシーケンシャル処理に比べ，5分程処理時間が長くなっている。

表1　シーケンシャル処理とパラレル処理

処理の種類	処理時間(s)
シーケンシャル	1323
パラレル	1647

これは，パラレル処理では3つのタスクが時分割で処理されるため，シーケンシャル処理にはない2つのタスクの処理にプロセッサの処理能力が割かれ，その結果Calculateタスクには80%の能力しか使用できなかった事を示している。

次に、使用するプロセッサの台数を変化させ，処理時間（単位は秒）を求めた。また，マルチプロセッサシステムの性能を記述する一般的な指標として用いられる「速度向上比S_p」と「効率E_p」[5]を計算し，あわせて表2に示す。

表2　プロセッサ数に対する処理時間

台数	処理時間	速度向上比	効率
1	1647	1.00	1.00
2	917	1.80	0.90
3	633	2.60	0.87
4	493	3.34	0.84
5	402	4.10	0.82
6	347	4.75	0.79
7	295	5.58	0.80
8	269	6.12	0.77
9	238	6.92	0.77

速度向上比は（1）式のように定義される。

$$S_p = \frac{T_1}{T_p} \qquad (1)$$

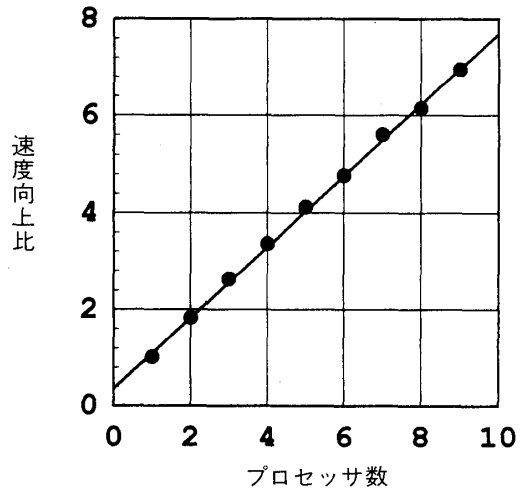
ここで，S_pはp個のノード（要素を接続する点；ここではトランスピュータの台数に相当する）での速度向上比を表す。T_1が単一ノードで問題を解くのに要する時間を表し，T_pがp個のノードで同じ問題を解くのに要する時間を表している。

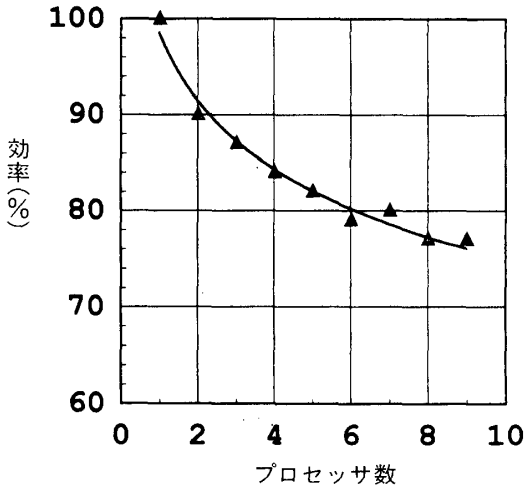
また，効率は（2）式のように定義され，この値は1となるのが理想的である。

$$E_p = \frac{T_1}{p \cdot T_p} \qquad (2)$$

図7にネットワークを構成するプロセッサ数に対する速度向上比および効率の変化を示す。図に示されるように速度向上比はほぼ直線状に向上することが分かる。このように，直線的に増加するケースでは，プロセッサ間

図7　速度向上比および効率





通信による計算時間の損失が少ないと言われ[5]，高速なシリアルリンクを有するトランスピュータの優位性が確かめられている。また，この直線の傾きがネットワークシステム全体の計算効率を表すことが（1），（2）式から導かれ，直線回帰から値を求めると0.74となった。文献では90%を超える効率を有するシステムも報告されており[6]，本システム構成にはまだ改良の余地が残されていると考えられる。また，効率はプロセッサ数の増加に伴い徐々に減少してゆき，次第に上述の74%へと近づいていくことも示された。

次に各プロセッサへの負荷の分散情況を把握するため，トランスピュータを9台用い各トランスピュータにおける処理個数とCalculateタスクでの演算時間を調べた。結果を表3に示す。

表3　データ数と演算時間

	処理個数（個）	演算時間（秒）
TP0	6	208
TP1	3	217
TP2	3	209
TP3	4	211
TP4	6	230
TP5	3	213
TP6	7	233
TP7	5	233
TP8	3	209

各トランスピュータ上のCalculateタスクでは処理したデータ数に関係なく演算時間が200～230（s）とほぼ均等になっている。すなわち，本ネットワークシステムでは各プロセッサともidle状態が無く，効率的な仕事の分配が成し遂げられたものと思われる。しかしながら，システム全体としてのCalculateタスクにおける演算時間の合計は32分43秒となり，1個のトランスピュータ上でパラレル処理を行った場合の演算時間（27分27秒）に比べ大幅に増大している。これは，各トランスピュータとも3つのタスクが時分割に処理されるために，Calculateタスクではトランスピュータの持つ能力の80%しか用いられないことによるものと考えられる。

4．マルチプロセッサ化に対する考察と問題点

マルチプロセッサによるネットワークシステムを構築する際には，効率的なアルゴリズムを設計することは最も重要であるが，

1）処理プロセッサ型とメモリ
2）各プロセッサへの負荷の分散
3）プロセッサの接続性

などを考慮すべきである。

本システムでは同一のプロセッサを用いたが，実際には経済的な見地から各プロセッサに分担させる仕事量に応じ，それに見合った能力を有するプロセッサを選択することも必要となる。また，ネットワーク内の全てのプロセッサが遊び時間なく仕事を繰り返すとき，システムは最高速度で処理を行うことができる。本報告でも，これを最重点に，ハードウェアとしてパイプライン接続およびソフトウェアとしてプロセッサファーム技法を用いネットワークを構築し，処理プロセッサを最高で9個使用することにより速度向上比を7近くにまで高めることができた。この組み合わせでは，容易にプロセッサの台数を増やすことが出来るという柔軟性を持ったシステムが構築できる。しかしながら，データの入出力を監視するタスクが必要であるなど計算効率を落とす要因がいくつかあった。

トランスピュータは，MIMD型のネットワークシステムを構成するのに適切なマイクロプロセッサであることが示された。しかしながら，現在のトランスピュータT800型ではリ

ンク数が固定されているため，何らかのスイッチングデバイスを用いない限り，多様な相互結合性を有するネットワークは構成できない。今回は，遠方のプロセッサにデータを送るためにいわゆる「バケツリレー型」の通信アーキテクチャを採用せざるを得なかった。その結果，プログラムが複雑かつ通信処理の効率化のネックとなっている。最近発表された上位機種T9000型[7]には，「バーチャル・リンク」という考え方が導入され，任意のプロセッサ間の通信が可能となっており，大きな期待が寄せられている。

5．おわりに

本報告ではグループウェアへの適用を目指し，トランスピュータを用いたコンピュータネットワークシステムをプロセッサファームの技法で構築し，速度向上比，効率および負荷分散などからシステムの特性を評価した。実際のLANで接続されたネットワークシステムへの拡張のためには，UNIXやMS-DOSで提供される「ソケット」というTCP/IPによる通信機能を用いることで可能と思われる。しかしながら，ネットワークには様々なワークステーションが含まれ，プロセッサの処理能力が均一でないため各機器の性能に見合ったタスクを配慮しなければならない，また高速な通信ができない等，新たな問題が生じてくるものと思われる。

これまでの様に単独のコンピュータでソフトウェア作成を行うときには，アルゴリズムの開発のみで良かったが，ネットワーク・コンピューティングを利用する際には，ハードウェア設計と専用のアルゴリズム開発という二次元的な試行が必要となり，コンピューティングが新しい局面を向かえようとしているものと思われる。

「参考文献」

1）石井裕：「グループウェアのデザイン」(1994)，共立出版

2）Ronald S.Cok，梅尾博司 監約：「並列プログラミング入門」(1993)，共立出版

3）山本正樹，中井泰明，村上安範：「トランスピュータ入門」(1990)，日刊工業新聞

4）“Parallel Fortran User Guide”(1988)，3L Ltd.

5）電気学会技術報告（II部），「気体シミュレーション技法」(1983)，電気学会

5）矢川元基 編：「コンピュートロール」40 (1992)，コロナ社

6）矢川元基，曾根田直樹：「パラレルコンピューティング」(1991)，培風館

7）IMS The T9000 Transputer Products Overview Manual (1991)，SGS-THOMSON